



A General Method for Representing Sets of Relations by Vectors

Rudolf Berghammer¹ and Michael Winter²(✉)

¹ Institut für Informatik, Universität Kiel, 24098 Kiel, Germany

² Department of Computer Science, Brock University, St. Catharines, ON, Canada
mwinter@brocku.ca

Abstract. We present a general method for computing vector representations \mathfrak{r} of sets of relations. This method is used for obtaining \mathfrak{r} from an inclusion $\mathfrak{R} \subseteq \mathfrak{S}$, where \mathfrak{R} and \mathfrak{S} are relation-algebraic expressions over a relation rather than its vector representation. The core of the method is a theorem that shows how \mathfrak{r} can be obtained from $\mathfrak{R} \subseteq \mathfrak{S}$ in one step. As applications we consider some problems concerning kernels of relations.

1 Introduction

Reduced ordered binary decision diagrams (ROBDDs) are a very efficient data structure for representing sets and relations. This has been shown by numerous applications of RELVIEW, a ROBDD-based tool for the manipulation and visualization of relations and relational programming (see [1]). The use of ROBDDs often leads to an amazing computational power, in particular, if a hard problem is solved by computing a very large set of ‘interesting objects’ or ‘potential solutions’ and subsequently selecting an (optimal) solution among them.

In many applications the set of potential solutions is a subset \mathcal{R} of the power set 2^X for some set X . A method for solving such a task is to start with a logical formula $\varphi(Y)$ that characterizes whether a set $Y \in 2^X$ belongs to \mathcal{R} or not. This formula is then transformed (using correspondences between logical and relation-algebraic constructions) into the form \mathfrak{r}_Y , where \mathfrak{r} is a relation-algebraic expression that evaluates to a relational vector in the sense of [10]. Finally, \mathfrak{r} is translated into the programming language of RELVIEW for execution. For some typical examples, see [2, 3, 5]. When Y is represented by a vector v , then very often $\varphi(Y)$ is equivalent to an inclusion $\mathfrak{v} \subseteq \mathfrak{w}$, where \mathfrak{v} and \mathfrak{w} are relation-algebraic expressions over v with a close or simple relationship between the inclusion and the expression \mathfrak{r} . Based on this observation, in [4] a general method was developed that allows obtaining the expression \mathfrak{r} directly from the inclusion $\mathfrak{v} \subseteq \mathfrak{w}$ if \mathfrak{v} and \mathfrak{w} are of a specific syntactic form, called column-wise extendible vector expressions. Normally this leads to a substantial simplification of the development of \mathfrak{r} .

M. Winter—The author gratefully acknowledges support from the Natural Sciences and Engineering Research Council of Canada (283267).

In certain situations X is a Cartesian product $X_1 \times X_2$ which means that \mathcal{R} is a subset of the set of relations with source X_1 and target X_2 . Examples can be found in [5, 7], where the original logic-based development method is applied to get \mathfrak{r} , and in [4, 6], where the method of [4] is used instead. In particular the examples of [6] demonstrate the superiority of the method of [4] with regard to the logic-based method. However, the method of [4] has an obvious disadvantage. It requires that both sides of the inclusion $\mathfrak{v} \subseteq \mathfrak{w}$ describing the property $R \in \mathcal{R}$ are relation-algebraic expressions over a vector representation of R and not over R itself, which is usually given and much simpler.

In this paper we present a new and very general method for computing vector representations \mathfrak{r} of sets \mathcal{R} of relations. This overcomes the just mentioned problem, i.e., it can be used for computing \mathfrak{r} from an inclusion $\mathfrak{R} \subseteq \mathfrak{S}$, where both sides are relation-algebraic expressions over a relation R and not over a vector representation of R . Decisive for this method is an equation that reduces the vector representation of a composition $R; S$ of relations to those of R and S . Together with known results concerning the remaining relation-algebraic operations, this allows the computation of vector representations via a recursive function ν_r and the proof of a theorem that shows how \mathfrak{r} can be obtained from $\mathfrak{R} \subseteq \mathfrak{S}$ directly. As applications we treat some of the problems of [7] with the new method.

2 Relational Preliminaries

In this section we want to recall some basic facts about (binary, set-theoretic) relations and their operations that are used throughout this paper. For more details on relation algebras, see [10, 11] for example.

Given sets X and Y , the power set $2^{X \times Y}$ of $X \times Y$ is the set of relations with *source* X and *target* Y , which we will denote by $[X \leftrightarrow Y]$. We will write $R : X \leftrightarrow Y$ instead of $R \in [X \leftrightarrow Y]$ and call $X \leftrightarrow Y$ the *type* of R . If X and Y are finite, then we may represent any relation $R : X \leftrightarrow Y$ by a Boolean matrix, in which an entry ‘true’ (or 1) in the row corresponding to $x \in X$ and in the column corresponding to $y \in Y$ indicates that $(x, y) \in R$. Analogously, the entry ‘false’ (or 0) indicates that the elements are not in relation R , i.e., $(x, y) \notin R$. This matrix interpretation is also used by the RELVIEW system to visualize relations. In addition, we will often use Boolean matrix notation and terminology in the remainder of this paper. In particular, we write $R_{x,y}$ instead of $(x, y) \in R$. Then we speak of a point-wise notation.

The following basic operations on relations will be used: \overline{R} (*complement*), $R \cup S$ (*union*), $R \cap S$ (*intersection*), R^T (*transposition* or *converse*), and $R; S$ (*composition*). We assume that transposition and complementation bind stronger than composition and composition binds stronger than union and intersection. In addition, we have the constants \mathbf{O} (*empty relation*), \mathbf{L} (*universal relation*) and \mathbf{I} (*identity relation*). Note that these constants are polymorphic, e.g., there is an identity relation of type $X \leftrightarrow X$ for all X . We also will use $R \subseteq S$ to indicate that R is included in S . We assume that the reader is familiar with these concepts.

As derived operation we will use $\text{syq}(R, S) := \overline{R^\top}; \overline{S} \cap \overline{R^\top}; S$, the *symmetric quotient* of $R : X \leftrightarrow Y$ and $S : X \leftrightarrow Z$. It easily can be shown that $\text{syq}(R, S)$ has type $Y \leftrightarrow Z$ and for all $y \in Y$ and $z \in Z$ it holds $\text{syq}(R, S)_{y,z}$ iff for all $x \in X$ it holds $R_{x,y}$ iff $S_{x,z}$.

Besides the well-known lattice theoretic properties and basic properties of composition and transposition, i.e., $Q; (R \cup S) = Q; R \cup Q; S$, $Q; (R \cap S) \subseteq Q; R \cap Q; S$, $(R \cup S)^\top = R^\top \cup S^\top$, $(R \cap S)^\top = R^\top \cap S^\top$, $(R; S)^\top = S^\top; R^\top$, $(R^\top)^\top = R$ and $\overline{R^\top} = \overline{R}^\top$, relations also satisfy the so-called *modular inclusion* $Q; R \cap S \subseteq Q; (R \cap Q^\top; S)$, and the so-called *Schröder equivalences* $Q^\top; \overline{S} \subseteq \overline{R}$ iff $Q; R \subseteq S$ iff $\overline{S}; R^\top \subseteq \overline{Q}$. In the remainder of the paper we will use these properties without mentioning. Some additional properties are summarized in the following lemma. A proof can be found in [10, 11].

Lemma 2.1. *Let be $Q : X \leftrightarrow Y$, $R : X \leftrightarrow Z$, $S : Y \leftrightarrow Z$, and $T : X \leftrightarrow Z$. Then we have*

- (1) $(Q \cap R; L); S = Q; S \cap R; L$,
- (2) $(Q; S \cap T); L = (Q \cap T; S^\top); L$.

An important class of relations are given by maps (or functions). We call a relation $Q : X \leftrightarrow Y$ *univalent* (or a partial function) iff $Q^\top; Q \subseteq \mathbb{I}$, *total* iff $\mathbb{I} \subseteq Q; Q^\top$, *injective* iff Q^\top is univalent, *surjective* iff Q^\top is total, and a *map* iff Q is total and univalent. The following lemma collects some important properties of univalent relations.

Lemma 2.2. *Assume $f : X \leftrightarrow Y$ to be univalent. Furthermore, let be $Q, R : Y \leftrightarrow Z$, $S : W \leftrightarrow X$, $T : W \leftrightarrow Y$, and $U : X \leftrightarrow Y$. Then we have*

- (1) $f; (Q \cap R) = f; Q \cap f; R$,
- (2) $(S \cap T; f^\top); f = S; f \cap T$,
- (3) $f \cap (f \cap U); L = f \cap U$.

Another important concept is the notion of pairs and the *projection relations* $\pi : X \times Y \leftrightarrow X$ and $\rho : X \times Y \leftrightarrow Y$. The projection relations have the Cartesian product $X \times Y$ as source and X resp. Y as target and are defined by $\pi_{(u_1, u_2), x}$ iff $u_1 = x$ and $\rho_{(u_1, u_2), y}$ iff $u_2 = y$, for all $(u_1, u_2) \in X \times Y$, $x \in X$ and $y \in Y$. These relations and the corresponding object $X \times Y$ can also be defined abstractly (up to isomorphism) by the formulas $\pi^\top; \pi \subseteq \mathbb{I}$, $\rho^\top; \rho \subseteq \mathbb{I}$, $\pi; \pi^\top \cap \rho; \rho^\top = \mathbb{I}$ and $\pi^\top; \rho = \mathbb{L}$, see [13]¹. To enhance presentation, in the remainder of this paper we will overload the projection relations, i.e., consider them as polymorphic. In all such cases it is easy to determine their types from the context using the typing rules of the operations of relation algebra.

¹ In [10, 11] instead of inclusions the first two axioms are equations. This leads to difficulties if precisely one of the sets of $X \times Y$ is empty. If e.g., $X \neq \emptyset$, then $\pi : X \times Y \leftrightarrow X$ is not surjective, whereas $\pi^\top; \pi = \mathbb{I}$ implies surjectivity. The weaker notion used here already implies the uniqueness of the construction up to isomorphism.

Based upon the projection relations we can define the *left pairing* of two relations $R : X \leftrightarrow Z$ and $S : Y \leftrightarrow Z$ by $\llbracket R, S \rrbracket := \pi; R \cap \rho; S$ of type $X \times Y \leftrightarrow Z$. When using a point-wise notation we have $\llbracket R, S \rrbracket_{(u_1, u_2), z}$ iff $R_{u_1, z}$ and $S_{u_2, z}$, for all $(u_1, u_2) \in X \times Y$ and $z \in Z$. Similar to the left pairing we can define the *right pairing* of two relations $R : Z \leftrightarrow X$ and $S : Z \leftrightarrow Y$ by $\llbracket R, S \rrbracket := R; \pi^\top \cap S; \rho^\top$. Here we obtain $Z \leftrightarrow X \times Y$ as type and that $\llbracket R, S \rrbracket_{z, (u_1, u_2)}$ iff R_{z, u_1} and S_{z, u_2} , for all $(u_1, u_2) \in X \times Y$ and $z \in Z$. Finally, the *parallel composition* (or *product*) $R \parallel S : X \times X \leftrightarrow Y \times Y'$ of two relations $R : X \leftrightarrow Y$ and $S : X' \leftrightarrow Y'$ is defined by $R \parallel S := \pi; R; \pi^\top \cap \rho; S; \rho^\top$, i.e., we have $(R \parallel S)_{(u_1, u_2), (v_1, v_2)}$ iff R_{u_1, v_1} and S_{u_2, v_2} , for all $(u_1, u_2) \in X \times X'$ and $(v_1, v_2) \in Y \times Y'$. As a consequence we have $R \parallel S = [\pi; R; \rho; S] = \llbracket R; \pi^\top, S; \rho^\top \rrbracket$, where the right pairing is formed w.r.t. the projection relations of $Y \times Y'$ and the left pairing is formed w.r.t. the projection relations of $X \times X'$.

From Lemma 2.2(2) we obtain

$$[Q, R]; \pi = (Q; \pi^\top \cap R; \rho^\top); \pi = Q \cap R; \rho^\top; \pi = Q \cap R; L,$$

so that $[Q, R]; \pi = Q$ follows if R is total. Analogously, we get $[Q, R]; \rho = R$ if Q is total and similar results for the left pairing and the parallel composition.

The sharpness property of relational products is the question whether the following equation

$$[Q, R]; \llbracket S, T \rrbracket = Q; S \cap R; T$$

holds for all relations Q, R, S and T of suitable types. Note that the equation only involves one Cartesian product, and is easy to verify for set-theoretic relations. However, the equation does not follow from the axioms of a relation algebra and of the corresponding projection relations alone. But if we require sufficient additional structure, i.e., the existence of at least one additional Cartesian product, we are able to show sharpness. The lemma below generalizes the equation above slightly. If instantiated with $S_1 = \pi$ and $S_2 = \rho$ it verifies sharpness under the assumption of the existence of the additional Cartesian product $W \times X$. The lemma itself is an immediate consequence of the approach developed in [8].

Lemma 2.3. *Let be $Q_1 : W \leftrightarrow X$, $Q_2 : W \leftrightarrow Y$, $R_1 : X \leftrightarrow Z$, $R_2 : Y \leftrightarrow Z$, $S_1 : V \leftrightarrow X$, and $S_2 : V \leftrightarrow Y$ with S_1 and S_2 univalent and*

$$Q_1^\top; Q_2 \cap R_1^\top; R_2 \subseteq S_1^\top; S_2.$$

Furthermore, assume that the Cartesian product $W \times X$ exists. Then we have

$$(Q_1; S_1^\top \cap Q_2; S_2^\top); (S_1; R_1^\top \cap S_2; R_2^\top) = Q_1; R_1^\top \cap Q_2; R_2^\top.$$

In the remainder of this paper we will assume that the Cartesian product for every pair of sets (objects) exists, together with the corresponding projection relations, such that we will always have sharpness.

The relation $S^{X, X'} : X \times X' \leftrightarrow X' \times X$ is defined by $S^{X, X'} := [\rho, \pi] = \llbracket \rho^\top, \pi^\top \rrbracket$. As usual we will drop the sets X and X' and write simply S instead of $S^{X, X'}$ if the sets of the Cartesian products are clear from the context.

The relation exchanges the two components of a pair, i.e., we get $S_{(u_1, u_2), (v_1, v_2)}$ iff $u_1 = v_2$ and $u_2 = v_1$, for all $(u_1, u_2) \in X \times X'$ and $(v_1, v_2) \in X' \times X$. Furthermore, we have $S^\top = S$ and $S; S = \mathbf{l}$, where the two occurrences of S in both equations are different versions of the polymorphic relation.

The partial identity $\mathbf{l} \cap \pi; \rho; \pi^\top; \rho^\top$ has source and target $(X \times Y) \times (Y \times Z)$. From this type information we can infer that the first occurrence of π denotes the first projection relation of the Cartesian product $(X \times Y) \times (Y \times Z)$, whereas the second occurrence of π denotes the first projection relation of the Cartesian product $Y \times Z$, and the first occurrence of ρ denotes the second projection relation of $X \times Y$, whereas the second occurrence of ρ denotes the second projection relation of $(X \times Y) \times (Y \times Z)$. The relation $\mathbf{l} \cap \pi; \rho; \pi^\top; \rho^\top$ acts as a filter when composing it with a suitable relation. Using point-wise notation, a quadruple $((u_1, u_2), (v_1, v_2)) \in (X \times Y) \times (Y \times Z)$ is related to itself by the relation $\mathbf{l} \cap \pi; \rho; \pi^\top; \rho^\top$ iff $u_2 = v_1$, representing the condition under with the pair $(u_1, v_2) \in X \times Z$ would be in the composition of two relations, where the first relation contains the pair $(u_1, u_2) \in X \times Y$ and the second relation contains the pair $(v_1, v_2) \in Y \times Z$. Note that we have

$$\mathbf{l} \cap \pi; \rho; \pi^\top; \rho^\top = (\mathbf{l} \cap \pi; \rho; \pi^\top; \rho^\top)^\top = \mathbf{l} \cap \rho; \pi; \rho^\top; \pi^\top.$$

By means of the partial identity $\mathbf{l} \cap \pi; \rho; \pi^\top; \rho^\top$ we now define the relation $C^{X, Y, Z}$ of type $(X \times Y) \times (Y \times Z) \leftrightarrow X \times Z$ by $C^{X, Y, Z} := (\mathbf{l} \cap \pi; \rho; \pi^\top; \rho^\top); (\pi \parallel \rho)$. This relation removes the common intermediate element. Again, we will usually drop the three sets X, Y and Z of $C^{X, Y, Z}$ and overload the relation.

In the next sections we also will use the relation-level equivalents of the set-theoretic symbol ‘ \in ’ as basic relations. These are the (again polymorphic) *membership relations* $M : X \leftrightarrow 2^X$, which are point-wisely described by $M_{x, Y}$ iff $x \in Y$, for all elements $x \in X$ and sets $Y \in 2^X$. There exists a relation-algebraic axiomatization of membership relations which specifies these up to isomorphism. See [11], for example. But for the applications of the present paper the above point-wise description suffices.

3 Vector Representation of Relations

Relational vectors are relations $v : X \leftrightarrow Y$ with $v; \mathbf{l} = v$. Such a v can be interpreted as a subset of X in the following sense: If represented by a Boolean matrix, the relation v is a matrix in which every row consists completely either of 1’s (or ‘true’-entries) or of 0’s (or ‘false’-entries) indicating that the element corresponding to that row either belongs to the subset of X or not. Since Y is irrelevant in this representation we will always consider vectors with target $\mathbf{1}$, where $\mathbf{1} = \{\perp\}$ is a specific singleton set. In this case all relations in the set $[X \leftrightarrow \mathbf{1}]$ of relations are vectors since for a singleton set we have $\mathbf{l} = \mathbf{l}$. Note that a singleton set can also be defined abstractly (up to isomorphism) as a so-called unit. A unit is an object $\mathbf{1}$ such that $\mathbf{l} = \mathbf{l}$ for $\mathbf{l} : \mathbf{1} \leftrightarrow \mathbf{1}$ and $\mathbf{l} : X \leftrightarrow \mathbf{1}$ is total for every X .

Relations are specific sets. In the following we concentrate on their representations by means of vectors.

The first step in providing a vector representation of arbitrary relations is to establish a Boolean lattice isomorphism between the set of relations $R : X \leftrightarrow Y$ and the set of vectors $v : X \times Y \leftrightarrow \mathbf{1}$. Given a relation $R : X \leftrightarrow Y$ and a vector $v : X \times Y \leftrightarrow \mathbf{1}$ we define $\text{vec}(R) : X \times Y \leftrightarrow \mathbf{1}$ and $\text{Rel}(v) : X \leftrightarrow Y$ as shown in the following figure (Fig. 1).

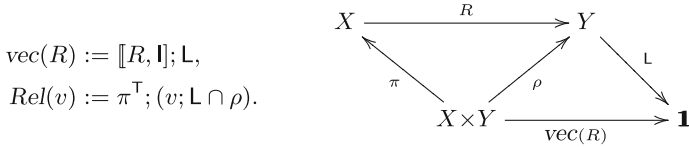


Fig. 1. Definition of functions vec and Rel

The functions $\text{vec} : [X \leftrightarrow Y] \rightarrow [X \times Y \leftrightarrow \mathbf{1}]$ and $\text{Rel} : [X \times Y \leftrightarrow \mathbf{1}] \rightarrow [X \leftrightarrow Y]$ are inverse to each other, i.e., we have $\text{Rel}(\text{vec}(R)) = R$ and $\text{vec}(\text{Rel}(v)) = v$, for all $R : X \leftrightarrow Y$ and $v : X \times Y \leftrightarrow \mathbf{1}$. Furthermore, vec and Rel are monotone w.r.t. inclusion, i.e., $R \subseteq S$ implies $\text{vec}(R) \subseteq \text{vec}(S)$, for all $R, S : X \leftrightarrow Y$, and $v \subseteq w$ implies $\text{Rel}(v) \subseteq \text{Rel}(w)$, for all $v, w : X \times Y \leftrightarrow \mathbf{1}$. We even have $R \subseteq S$ iff $\text{vec}(R) \subseteq \text{vec}(S)$, for all $R, S : X \leftrightarrow Y$, and $v \subseteq w$ iff $\text{Rel}(v) \subseteq \text{Rel}(w)$, for all $v, w : X \times Y \leftrightarrow \mathbf{1}$. Proofs of these facts can be found in [10], for example.

The next lemma shows how the five basic operations of relation algebra can be performed directly on the corresponding vectors. Together with the bijectivity properties (1), (3) and (4) of Lemma 3.1 imply that the function vec is an isomorphism from the Boolean lattice $[X \leftrightarrow Y]$ of all relations between X and Y to the Boolean lattice $[X \times Y \leftrightarrow \mathbf{1}]$ of all vectors with source $X \times Y$, with $\text{vec}^{-1} = \text{Rel}$, such that $\text{vec}(\mathbf{O}) = \mathbf{O}$, $\text{vec}(\mathbf{L}) = \mathbf{L}$, $\text{Rel}(\mathbf{O}) = \mathbf{O}$ and $\text{Rel}(\mathbf{L}) = \mathbf{L}$ immediately follows.

Lemma 3.1. *Let be $Q, R : X \leftrightarrow Y$ and $S : Y \leftrightarrow Z$. Then we have*

- (1) $\text{vec}(\overline{Q}) = \overline{\text{vec}(Q)}$,
- (2) $\text{vec}(Q^\top) = \mathbf{S}; \text{vec}(Q)$,
- (3) $\text{vec}(Q \cup R) = \text{vec}(Q) \cup \text{vec}(R)$,
- (4) $\text{vec}(Q \cap R) = \text{vec}(Q) \cap \text{vec}(R)$,
- (5) $\text{vec}(R; S) = \mathbf{C}^\top; [\text{vec}(R), \text{vec}(S)]$.

Proof. The four properties (1) to (4) were already shown in [10]. In order to prove property (5), we would like to show that the composition $\mathbf{C}^\top; [\text{vec}(R), \text{vec}(S)]$ can basically be written as the left-hand side of the equation in Lemma 2.3 for suitable S_1 and S_2 , and then apply that lemma. Therefore, we will use the following abbreviations

$$\begin{aligned}
 i &:= \mathbf{I} \cap \pi; \rho; \pi^\top; \rho^\top, & \tilde{\pi} &:= \pi \cap \rho; \pi; \rho^\top, \\
 f &:= \pi; \rho \cap \rho; \pi, & \tilde{\rho} &:= \rho \cap \pi; \rho; \pi^\top.
 \end{aligned}$$

Obviously, all four relations are univalent and we have $i; \pi = \tilde{\pi}$, $i; \rho = \tilde{\rho}$, $\tilde{\pi}; \rho = f$ and $\tilde{\rho}; \pi = f$, where the last four equations follow immediately from Lemma 2.2(2). Furthermore, we have $\tilde{\pi}; \llbracket R, \mathbb{I} \rrbracket \subseteq \tilde{\pi}; \rho = f$ and

$$\begin{aligned}
\tilde{\rho}; \llbracket S, \mathbb{I} \rrbracket; \mathbb{L} \cap f &= \tilde{\rho}; (\pi; S \cap \rho); \mathbb{L} \cap f \\
&= \tilde{\rho}; (\pi \cap \rho; S^\top); \mathbb{L} \cap f && \text{Lemma 2.1(2)} \\
&= (\tilde{\rho}; \pi \cap \tilde{\rho}; \rho; S^\top); \mathbb{L} \cap f && \text{Lemma 2.2(1)} \\
&= (f \cap \tilde{\rho}; \rho; S^\top); \mathbb{L} \cap f \\
&= f \cap \tilde{\rho}; \rho; S^\top && \text{Lemma 2.2(3)} \\
&= \tilde{\rho}; \pi \cap \tilde{\rho}; \rho; S^\top \\
&= \tilde{\rho}; (\pi \cap \rho; S^\top) && \text{Lemma 2.2(1)} \\
&= \tilde{\rho}; \llbracket \mathbb{I}, S^\top \rrbracket.
\end{aligned}$$

These two properties imply

$$\begin{aligned}
i; \llbracket \text{vec}(R), \text{vec}(S) \rrbracket &= i; \pi; \llbracket R, \mathbb{I} \rrbracket; \mathbb{L} \cap i; \rho; \llbracket S, \mathbb{I} \rrbracket; \mathbb{L} && \text{Lemma 2.2(1)} \\
&= \tilde{\pi}; \llbracket R, \mathbb{I} \rrbracket; \mathbb{L} \cap \tilde{\rho}; \llbracket S, \mathbb{I} \rrbracket; \mathbb{L} \\
&= (\tilde{\pi}; \llbracket R, \mathbb{I} \rrbracket \cap \tilde{\rho}; \llbracket S, \mathbb{I} \rrbracket; \mathbb{L}); \mathbb{L} && \text{Lemma 2.1(1)} \\
&= (\tilde{\pi}; \llbracket R, \mathbb{I} \rrbracket \cap f \cap \tilde{\rho}; \llbracket S, \mathbb{I} \rrbracket; \mathbb{L}); \mathbb{L} && \text{see above} \\
&= (\tilde{\pi}; \llbracket R, \mathbb{I} \rrbracket \cap \tilde{\rho}; \llbracket \mathbb{I}, S^\top \rrbracket); \mathbb{L} && \text{see above.}
\end{aligned}$$

On the other hand, we have

$$\begin{aligned}
\mathbb{C} &= i; (\pi \parallel \rho) \\
&= i; (\pi; \pi; \pi^\top \cap \rho; \rho; \rho^\top) \\
&= i; \pi; \pi; \pi^\top \cap i; \rho; \rho; \rho^\top && \text{Lemma 2.2(1)} \\
&= \tilde{\pi}; \pi; \pi^\top \cap \tilde{\rho}; \rho; \rho^\top
\end{aligned}$$

so that the composition $\mathbb{C}^\top; \llbracket \text{vec}(R), \text{vec}(S) \rrbracket$ in fact can be written as the left-hand side of Lemma 2.3 with $S_1 = \tilde{\pi}$ and $S_2 = \tilde{\rho}$.

The following calculation now shows that the additional assumption of Lemma 2.3 is satisfied:

$$\begin{aligned}
& \llbracket R, \mathbb{I} \rrbracket; \llbracket \mathbb{I}, S^T \rrbracket^T \cap (\pi; \pi^T)^T; \rho; \rho^T \\
&= \llbracket R, \mathbb{I} \rrbracket; \llbracket \mathbb{I}, S^T \rrbracket^T \cap \pi; \pi^T; \rho; \rho^T \\
&= \llbracket R, \mathbb{I} \rrbracket; \llbracket \mathbb{I}, S^T \rrbracket^T \cap \pi; \mathbb{L}; \rho^T \\
&= \llbracket R, \mathbb{I} \rrbracket; \llbracket \mathbb{I}, S^T \rrbracket^T && \pi \text{ and } \rho \text{ total} \\
&\subseteq \rho; \pi^T \\
&= \rho; \pi^T \cap \rho; \pi^T \\
&= \llbracket \mathbb{I}, \rho; \pi^T \rrbracket; \llbracket \rho; \pi^T, \mathbb{I} \rrbracket && \text{sharpness} \\
&= (\pi^T \cap \rho; \pi^T; \rho^T); (\rho \cap \pi; \rho; \pi^T) \\
&= \tilde{\pi}^T; \tilde{\rho},
\end{aligned}$$

As a consequence we conclude

$$(\pi; \pi^T; \tilde{\pi}^T \cap \rho; \rho^T; \tilde{\rho}^T); (\tilde{\pi}; \llbracket R, \mathbb{I} \rrbracket \cap \tilde{\rho}; \llbracket \mathbb{I}, S^T \rrbracket) = \pi; \pi^T; \llbracket R, \mathbb{I} \rrbracket \cap \rho; \rho^T; \llbracket \mathbb{I}, S^T \rrbracket.$$

We finally obtain

$$\begin{aligned}
& \text{vec}(R; S) \\
&= \llbracket R; S, \mathbb{I} \rrbracket; \mathbb{L} \\
&= (\pi; R; S \cap \rho); \mathbb{L} \\
&= (\pi; R \cap \rho; S^T); \mathbb{L} && \text{Lemma 2.1(2)} \\
&= (\pi; \pi^T; \llbracket R, \mathbb{I} \rrbracket \cap \rho; \rho^T; \llbracket \mathbb{I}, S^T \rrbracket); \mathbb{L} \\
&= (\pi; \pi^T; \tilde{\pi}^T \cap \rho; \rho^T; \tilde{\rho}^T); (\tilde{\pi}; \llbracket R, \mathbb{I} \rrbracket \cap \tilde{\rho}; \llbracket \mathbb{I}, S^T \rrbracket); \mathbb{L} && \text{see above} \\
&= C^T; i; \llbracket \text{vec}(R), \text{vec}(S) \rrbracket \\
&= C^T; \llbracket \text{vec}(R), \text{vec}(S) \rrbracket, && i^T; i = i; i = i = i^T
\end{aligned}$$

i.e., the desired property (5). \square

The relation C allows us to express a composition of two relations as an operation on their corresponding vectors. Sometimes, when computing all relations that satisfy a certain property, it is sufficient to convert only one relation in a series of compositions into a vector. Quite often this even leads to a more efficient implementation (e.g., in RELVIEW) of testing the property. The following lemma was shown in [9] and provides exactly such a translation.

Lemma 3.2. *Let $Q : W \leftrightarrow X$, $R : X \leftrightarrow Y$, and $S : Y \leftrightarrow Z$. Then we have*

$$\text{vec}(Q; R; S) = (Q \parallel S^T); \text{vec}(R).$$

As specific cases we get $\text{vec}(Q; R) = (Q \parallel \mathbb{I}); \text{vec}(R)$ by taking S as identity relation and $\text{vec}(R; S) = (\mathbb{I} \parallel S^T); \text{vec}(R)$ by taking Q as identity relation.

4 Vector Representation of Sets of Relations

In [9] so-called vector predicates are introduced for the relational treatment of evolutionary algorithms. They are functions in the usual mathematical sense on relations and model those sets of relations which are built from vectors using as operations only complementation, union, intersection and a restricted version of composition. This approach is continued and refined in [4] in view of the specification of vectors $v : 2^X \leftrightarrow \mathbf{1}$ which represent subsets \mathcal{R} of given power sets 2^X . The aim is to avoid lengthy and complex logical calculations and to work rather with relation-algebraic specifications of the elements of \mathcal{R} via inclusions $\mathbf{v} \subseteq \mathbf{w}$ such that v can be obtained from $\mathbf{v} \subseteq \mathbf{w}$ in one step using a general procedure. Decisive for this is that both sides of the inclusions are so-called column-wise extendible vector expressions. These are specific relation-algebraic expressions which can be seen as syntactical counterpart of vector predicates. Formally, they are defined as follows.

Definition 4.1. *Given a variable v of type $X \leftrightarrow \mathbf{1}$, the set \mathbf{VE}_v of typed column-wise extendible vector expressions over v is inductively defined as follows:*

- (1) *We have $v \in \mathbf{VE}_v$ and its type is $X \leftrightarrow \mathbf{1}$.*
- (2) *If $w : Y \leftrightarrow \mathbf{1}$, then $w \in \mathbf{VE}_v$ and its type is $Y \leftrightarrow \mathbf{1}$.*
- (3) *If $\mathbf{v} \in \mathbf{VE}_v$ is of type $Y \leftrightarrow \mathbf{1}$, then $\overline{\mathbf{v}} \in \mathbf{VE}_v$ and its type is $Y \leftrightarrow \mathbf{1}$.*
- (4) *If $\mathbf{v}, \mathbf{w} \in \mathbf{VE}_v$ are of type $Y \leftrightarrow \mathbf{1}$, then $\mathbf{v} \cup \mathbf{w} \in \mathbf{VE}_v$ and $\mathbf{v} \cap \mathbf{w} \in \mathbf{VE}_v$ and their types are $Y \leftrightarrow \mathbf{1}$.*
- (5) *If $\mathbf{v} \in \mathbf{VE}_v$ is of type $Y \leftrightarrow \mathbf{1}$ and \mathfrak{R} is a relation-algebraic expression of type $Z \leftrightarrow Y$ in which v does not occur, then $\mathfrak{R}; \mathbf{v} \in \mathbf{VE}_s$ and its type is $Z \leftrightarrow \mathbf{1}$.*

So, the vector expressions from \mathbf{VE}_v are built from the variable v using vectors and as operations only complementation, union, intersection and left-composition with a relation-algebraic expression in which v does not occur. Note that v is the only variable in such an expression. In the following we also allow the use of derived operations like symmetric quotients and pairings, but these are only seen as abbreviations. For example, $R; \llbracket v, S; v \rrbracket$ is considered as a column-wise extendible vector expression over the variable v since unfolding the definition of the left pairing yields $R; (\pi; v \cap \rho; S; v)$.

In a column-wise extendible vector expression \mathbf{v} over v the variable v can be replaced by a relation R with the same source as v . The result is denoted as $\mathbf{v}[R/v]$ and is inductively defined as follows.

Definition 4.2. *Given a variable v of type $X \leftrightarrow \mathbf{1}$, $\mathbf{v} \in \mathbf{VE}_v$ and $R : X \leftrightarrow Z$, we define $\mathbf{v}[R/v]$ as follows, using induction on the structure of \mathbf{v} :*

- (1) $v[R/v] = R$.
- (2) $w[R/v] = w; \mathbf{L}$, with $\mathbf{L} : \mathbf{1} \leftrightarrow Z$.
- (3) $\overline{\mathbf{v}}[R/v] = \overline{\mathbf{v}[R/v]}$.
- (4) $(\mathbf{v} \cup \mathbf{u})[R/v] = \mathbf{v}[R/v] \cup \mathbf{u}[R/v]$ and $(\mathbf{v} \cap \mathbf{u})[R/v] = \mathbf{v}[R/v] \cap \mathbf{u}[R/v]$.
- (5) $(\mathfrak{R}; \mathbf{v})[R/v] = \mathfrak{R}; (\mathbf{v}[R/v])$.

For example, for the variable v of type $X \leftrightarrow \mathbf{1}$, $S : Y \leftrightarrow X$ and $w : Y \leftrightarrow \mathbf{1}$ we have $S; v \cap w \in \mathbf{VE}_v$. A replacement of v in $S; v \cap w$ by the membership relation $M : X \leftrightarrow 2^X$ then yields $S; M \cap w; L$, which has type $Y \leftrightarrow 2^X$.

The general procedure to obtain the vectors $v : 2^X \leftrightarrow \mathbf{1}$ from the inclusions $\mathfrak{v} \subseteq \mathfrak{w}$, that we have mentioned above, is shown in [4] as Theorem 1. If we instantiate this theorem in such a way that instead of a general power set 2^X a power set of a Cartesian product $X \times Y$ is taken, i.e., a vector $v : [X \leftrightarrow Y] \leftrightarrow \mathbf{1}$ representing a subset \mathcal{R} of the set $[X \leftrightarrow Y]$ of relations is to be computed, then we get the following result.

Theorem 4.1. *Let the subset \mathcal{R} of the set $[X \leftrightarrow Y]$ of relations be specified as $\mathcal{R} = \{Rel(r) \mid r \in [X \times Y \leftrightarrow \mathbf{1}] \wedge \mathfrak{v} \subseteq \mathfrak{w}\}$, where $\mathfrak{v}, \mathfrak{w} \in \mathbf{VE}_r$. Using the membership relation $M : X \times Y \leftrightarrow [X \leftrightarrow Y]$ the set \mathcal{R} is represented by the vector*

$$v := \overline{L; (\mathfrak{v}[M/r] \cap \overline{\mathfrak{w}[M/r]})}^\top : [X \leftrightarrow Y] \leftrightarrow \mathbf{1}.$$

Using this result, in [6] a lot of vectors are obtained which represent important classes of relations. A certain disadvantage of Theorem 4.1 is that it works not directly with relations but with their vector representations. If, e.g., \mathcal{R} is the set of all transitive relations on a set X , then one would like to specify that R is a member of \mathcal{R} by $R; R \subseteq R$ instead of by $C^\top; \llbracket r, r \rrbracket \subseteq r$ as in [6], where r is the vector representation of R . Using the properties of the two functions vec and Rel given in Sect. 3, it is not hard to calculate the inclusion $C^\top; \llbracket r, r \rrbracket \subseteq r$ (between column-wise extendible vector expressions over r) from the common specification $Rel(r); Rel(r) \subseteq Rel(r)$ of $Rel(r)$ to be transitive.

In the following we generalize the example of transitive relations and consider arbitrary inclusions $\mathfrak{R} \subseteq \mathfrak{S}$, where \mathfrak{R} and \mathfrak{S} are relation-algebraic expressions that are constructed from a variable R of type $X \leftrightarrow Y$ using certain relations (e.g., membership relations and projection relations) and the constants and operations (including again also derived ones) of relation algebra. We denote the set of all these relation-algebraic expressions as \mathbf{RE}_R . Note that R is the only variable in such an expression. Our aim is to get from the specification $\mathcal{R} = \{R \in [X \leftrightarrow Y] \mid \mathfrak{R} \subseteq \mathfrak{S}\}$ a vector representation of this set in one step similar to Theorem 4.1. Decisive is the following function that transforms expressions from \mathbf{RE}_R into expressions from \mathbf{VE}_r , where $r : X \times Y \leftrightarrow \mathbf{1}$.

Definition 4.3. *Given a variable r of type $X \times Y \leftrightarrow \mathbf{1}$ and a variable R of type $X \leftrightarrow Y$, we define the function $\nu_r : \mathbf{RE}_R \rightarrow \mathbf{VE}_r$ as follows, using induction on the structure of the argument:*

- (1) $\nu_r(R) = r$.
- (2) $\nu_r(S) = vec(S)$ for all relations S .
- (3) $\nu_r(\mathfrak{R}) = \nu_r(\mathfrak{R})$.
- (4) $\nu_r(\mathfrak{R}^\top) = S; \nu_r(\mathfrak{R})$.
- (5) $\nu_r(\mathfrak{R} \cup \mathfrak{S}) = \nu_r(\mathfrak{R}) \cup \nu_r(\mathfrak{S})$.
- (6) $\nu_r(\mathfrak{R} \cap \mathfrak{S}) = \nu_r(\mathfrak{R}) \cap \nu_r(\mathfrak{S})$.
- (7) $\nu_r(\mathfrak{R}; \mathfrak{S}) = C^\top; \llbracket \nu_r(\mathfrak{R}), \nu_r(\mathfrak{S}) \rrbracket$.

In the next lemma we verify that ν_r in fact yields a column-wise extendible vector expression over the variable r . Furthermore, we show that if r is instantiated as $\text{vec}(R)$, then ν_r equals the function vec .

Lemma 4.1. *Let be a variable r of type $X \times Y \leftrightarrow \mathbf{1}$ and a variable R of type $X \leftrightarrow Y$. For all $\mathfrak{R} \in \mathbf{RE}_R$ we then have $\nu_r(\mathfrak{R}) \in \mathbf{VE}_r$ and, provided r is instantiated as $\text{vec}(R)$, also $\nu_r(\mathfrak{R}) = \text{vec}(\mathfrak{R})$.*

Proof. We use induction on the structure of the expression \mathfrak{R} .

The induction base is that \mathfrak{R} is the variable R or a relation S . In the first case we have $\nu_r(R) = r \in \mathbf{VE}_r$ and $\nu_r(R) = r$. So, if r is instantiated as $\text{vec}(R)$, then $\nu_r(\mathfrak{R}) = \text{vec}(\mathfrak{R})$. The case that \mathfrak{R} is a relation S is trivial.

In the first case of the induction step, assume \mathfrak{R} to be of the form $\overline{\mathfrak{S}}$ and that the induction hypothesis holds for the expression \mathfrak{S} . Then $\nu_r(\mathfrak{S}) \in \mathbf{VE}_r$ and Definition 4.3(3) yield $\nu_r(\overline{\mathfrak{S}}) \in \mathbf{VE}_r$. If r is instantiated as $\text{vec}(R)$, we have $\nu_r(\overline{\mathfrak{S}}) = \text{vec}(\overline{\mathfrak{S}})$ and we obtain

$$\nu_r(\overline{\mathfrak{S}}) = \overline{\nu_r(\mathfrak{S})} = \overline{\text{vec}(\mathfrak{S})} = \text{vec}(\overline{\mathfrak{S}})$$

using Definition 4.3(3) and Lemma 3.1(1). Next, let \mathfrak{R} be of the form \mathfrak{S}^\top and the induction hypothesis hold for the expression \mathfrak{S} . Here $\nu_r(\mathfrak{S}) \in \mathbf{VE}_r$ and Definition 4.3(4) show $\nu_r(\mathfrak{S}^\top) \in \mathbf{VE}_r$. As an instantiation of r as $\text{vec}(R)$ yields $\nu_r(\mathfrak{S}^\top) = \text{vec}(\mathfrak{S}^\top)$, we get

$$\nu_r(\mathfrak{S}^\top) = \mathfrak{S}; \nu_r(\mathfrak{S}) = \mathfrak{S}; \text{vec}(\mathfrak{S}^\top) = \text{vec}(\mathfrak{S}^\top),$$

using Definition 4.3(4) and Lemma 3.1(2). In the same way the cases can be treated where \mathfrak{R} is a union, an intersection and a composition of expressions. When \mathfrak{R} is a union, Definition 4.3(5) and Lemma 3.1(3) apply, when it is an intersection, Definition 4.3(6) and Lemma 3.1(4) apply, and when it is a composition, Definition 4.3(7) and Lemma 3.1(5) apply. \square

Now, we are able to prove the following variant of Theorem 4.1. The main difference in this modified version is that the elements of the set of relations are specified by an inclusion of relation-algebraic expressions over the variable R rather than as an inclusion between column-wise extendible vector expressions over a variable r that stands for the vector representation of R .

Theorem 4.2. *Let the subset \mathcal{R} of the set $[X \leftrightarrow Y]$ of relations be specified as $\mathcal{R} = \{R \in [X \leftrightarrow Y] \mid \mathfrak{R} \subseteq \mathfrak{S}\}$, where $\mathfrak{R}, \mathfrak{S} \in \mathbf{RE}_R$. Taking a variable r of type $X \times Y \leftrightarrow \mathbf{1}$ and using the membership relation $\mathbf{M} : X \times Y \leftrightarrow [X \leftrightarrow Y]$, the set \mathcal{R} is represented by the vector*

$$v := \overline{\overline{\nu_r(\mathfrak{R})[\mathbf{M}/r] \cap \overline{\nu_r(\mathfrak{S})[\mathbf{M}/r]}}^\top} : [X \leftrightarrow Y] \leftrightarrow \mathbf{1}.$$

Proof. First, we prove that the original specification of the set \mathcal{R} is equivalent to the specification

$$\mathcal{R} = \{\text{Rel}(r) \mid r \in [X \times Y \leftrightarrow \mathbf{1}] \wedge \nu_r(\mathfrak{R}) \subseteq \nu_r(\mathfrak{S})\}.$$

Let $R : X \leftrightarrow Y$ be given. We have to verify that $\mathfrak{R} \subseteq \mathfrak{S}$ holds iff there exists $r : X \times Y \leftrightarrow \mathbf{1}$ such that $\nu_r(\mathfrak{R}) \subseteq \nu_r(\mathfrak{S})$ and $R = \text{Rel}(r)$. Here is the proof, where r ranges over $[X \times Y \leftrightarrow \mathbf{1}]$:

$$\begin{aligned}
\mathfrak{R} \subseteq \mathfrak{S} &\iff \text{vec}(\mathfrak{R}) \subseteq \text{vec}(\mathfrak{S}) && \text{see Sect. 3} \\
&\iff \nu_{\text{vec}(R)}(\mathfrak{R}) \subseteq \nu_{\text{vec}(R)}(\mathfrak{S}) && \text{by Lemma 4.1} \\
&\iff \exists r : r = \text{vec}(R) \wedge \nu_r(\mathfrak{R}) \subseteq \nu_r(\mathfrak{S}) \\
&\iff \exists r : R = \text{Rel}(r) \wedge \nu_r(\mathfrak{R}) \subseteq \nu_r(\mathfrak{S}) && \text{see Sect. 3}
\end{aligned}$$

From Lemma 4.1 we also get that both sides of the inclusion of the second specification of \mathcal{R} are column-wise extendible vector expressions over r . Hence, Theorem 4.1 is applicable and yields the desired result. \square

As an example for applying the previous theorem, we consider again transitive relations, i.e., we use the specification

$$\mathcal{R} = \{R \in [X \leftrightarrow X] \mid R; R \subseteq R\}.$$

For arbitrary $R : X \leftrightarrow X$ and $r : X \times X \leftrightarrow \mathbf{1}$ we get for the left-hand side of the inclusion $R; R \subseteq R$ that $\nu_r(R; R) = C^\top; \llbracket \nu_r(R), \nu_r(R) \rrbracket = C^\top; \llbracket r, r \rrbracket$, such that $\nu_r(R; R)[M/r] = C^\top; \llbracket M, M \rrbracket$, with $M : X \times X \leftrightarrow [X \leftrightarrow X]$ as membership relation. In case of the right-hand side of $R; R \subseteq R$ we have $\nu_r(R) = r$, and this yields $\nu_r(R)[M/r] = M$. So, Theorem 4.2 implies that the vector

$$\text{trans} := \overline{C^\top; \llbracket M, M \rrbracket \cap \overline{M}}^\top$$

of type $[X \leftrightarrow X] \leftrightarrow \mathbf{1}$ represents \mathcal{R} , i.e., the set of transitive relations on X .

5 Applications: Kernels and Richardson's Criterion

A directed simple graph, i.e., a directed graph without multiple edges between the same pair of vertices, on a set of vertices X can be represented by a relation $R : X \leftrightarrow X$. A subset K of X is a *kernel* of R if for all $x \in X$ it holds $x \notin K$ iff there exists $y \in K$ such that $R_{x,y}$. Kernels have been introduced in [12] as a generalization of a solution of a cooperative game. Not every relation has a kernel and it is known that determining whether a relation has a kernel is a NP-complete problem.

There exist a series of sufficient criteria for the existence of kernels which can be tested efficiently. In [7] it was investigated how well these criteria characterize the class of relations that have kernels. This was done by computing the number of relations with a kernel for all sets X up to 7 elements using RELVIEW. Then the four most popular criteria of the above mentioned series were considered and for each of them the number of relations satisfying the criteria was computed – again for all X with $|X| \leq 7$ using RELVIEW. The numerical data of [7] show that even in case of the most general of the criteria, the absence of odd cycles

(Richardson's criterion), only a very small portion of the relations with kernels satisfy this property. So, the criteria are very far away from characterizing the class of relations with kernels. The data also led to the following conjecture for finite sets X : The probability that a relation selected uniformly at random from $[X \leftrightarrow X]$ has a kernel tends to zero if $|X|$ tends to infinity.

The key for getting the data of [7] are relation-algebraic specifications of five vectors of type $[X \leftrightarrow X] \leftrightarrow \mathbf{1}$ which represent the set of relations on X having kernels and the set of relations on X satisfying one of the criteria. The RELVIEW-programs then are nothing else than translations of the specifications into the programming language of the system. Each vector is obtained from a logical description of the relations in question and its transformation into an equivalent relation-algebraic expression. Especially in case of Richardson's criterion the development is rather technical and complex. In the following we solve two of the problems of [7] using the new method. A comparison with [7] shows that the new solutions are much more simple and many steps are very straight-forward.

We start with the characterization of relations having kernels. Assume $R : X \leftrightarrow X$ and $K \subseteq X$. Using the definition of a kernel and the point-wise description of symmetric quotients, we have that K is a kernel of R iff $\text{syq}(\overline{M}, R; M)_{K, K}$, where $M : X \leftrightarrow 2^X$ is a membership relation. As a consequence R has a kernel iff $\text{syq}(\overline{M}, R; M) \cap \mathbf{1} \neq \mathbf{0}$. Thus, R has no kernel iff $\text{syq}(\overline{M}, R; M) \subseteq \bar{\mathbf{1}}$. In order to obtain a vector *kernel* : $[X \leftrightarrow X] \leftrightarrow \mathbf{1}$ that represents the set of relations on X with a kernel, we use our method to transform the inclusion $\text{syq}(\overline{M}, R; M) \subseteq \bar{\mathbf{1}}$ into a vector *nokernel* : $[X \leftrightarrow X] \leftrightarrow \mathbf{1}$ and define *kernel* as complement of *nokernel*.

For the development of the vector *nokernel*, assume an arbitrary $r : X \times X \leftrightarrow \mathbf{1}$ to be given. Using the definition of the symmetric quotient, the definition of the function ν_r and Lemma 3.1 in combination with the abbreviations $m := \text{vec}(M)$ and $mt := \text{vec}(M^T)$, we get

$$\begin{aligned}
& \nu_r(\text{syq}(\overline{M}, R; M)) \\
&= \nu_r(\overline{\overline{M^T}; R; M} \cap \overline{\overline{M^T}; R; M}) \\
&= \nu_r(\overline{\overline{M^T}; R; M}) \cap \nu_r(\overline{\overline{M^T}; R; M}) \\
&= \nu_r(\overline{\overline{M^T}; R; M}) \cap \nu_r(\overline{\overline{M^T}; R; M}) \\
&= \overline{C^T; [\nu_r(\overline{M^T}), \nu_r(\overline{R; M})]} \cap \overline{C^T; [\nu_r(\overline{M^T}), \nu_r(\overline{R; M})]} \\
&= \overline{C^T; [\text{vec}(\overline{M^T}), \nu_r(\overline{R; M})]} \cap \overline{C^T; [\text{vec}(\overline{M^T}), \nu_r(\overline{R; M})]} \\
&= \overline{C^T; [\text{vec}(\overline{M^T}), C^T; [\nu_r(\overline{R}), \nu_r(\overline{M})]]} \cap \overline{C^T; [\text{vec}(\overline{M^T}), C^T; [\nu_r(\overline{R}), \nu_r(\overline{M})]]} \\
&= \overline{C^T; [\text{vec}(\overline{M^T}), C^T; [r, \text{vec}(M)]]} \cap \overline{C^T; [\text{vec}(\overline{M^T}), C^T; [r, \text{vec}(M)]]} \\
&= \overline{C^T; [\overline{mt}, C^T; [r, m]]} \cap \overline{C^T; [\overline{mt}, C^T; [r, m]]}
\end{aligned}$$

for the left-hand side of the inclusion $\text{syq}(\overline{\mathbf{M}}, R; \mathbf{M}) \subseteq \overline{\mathbf{I}}$, hence

$$\nu_r(\text{syq}(\overline{\mathbf{M}}, R; \mathbf{M}))[\mathbf{M}/r] = \overline{\mathbf{C}^\top; \overline{[mt; \overline{\mathbf{L}}, \mathbf{C}^\top; \overline{[\mathbf{M}, m; \mathbf{L}]}]} \cap \mathbf{C}^\top; \overline{[mt; \mathbf{L}, \mathbf{C}^\top; \overline{[\mathbf{M}, m; \mathbf{L}]}]}}$$

for the replacement of r by the membership relation $\mathbf{M} : X \times X \leftrightarrow [X \leftrightarrow X]$. In case of the right-hand side of the inclusion $\text{syq}(\overline{\mathbf{M}}, R; \mathbf{M}) \subseteq \overline{\mathbf{I}}$ we have $\nu_r(\overline{\mathbf{I}}) = \text{vec}(\overline{\mathbf{I}}) = \overline{\text{vec}(\mathbf{I})}$ and with the abbreviation $i := \text{vec}(\mathbf{I})$ we get

$$\nu_r(\overline{\mathbf{I}})[\mathbf{M}/r] = \overline{i}; \mathbf{L} = \overline{i}; \overline{\mathbf{L}}$$

for the replacement of r by \mathbf{M} . Now, Theorem 4.2 immediately yields

$$\text{nokernel} := \overline{\mathbf{L}; (\mathbf{C}^\top; \overline{[mt; \overline{\mathbf{L}}, \overline{\mathbf{H}}]} \cap \mathbf{C}^\top; \overline{[mt; \mathbf{L}, \overline{\mathbf{H}}]} \cap i; \mathbf{L})}^\top, \text{ where } \mathbf{H} := \mathbf{C}^\top; \overline{[\mathbf{M}, m; \mathbf{L}]}.$$

Next, we consider Richardson's criterion and assume, as in [7], a finite set X with $|X| \leq 7$. To compute a vector that represents the set of relations on X without odd cycles, in [7] four vectors cyc1 , cyc3 , cyc5 and cyc7 of type $[X \leftrightarrow X] \leftrightarrow \mathbf{1}$ were developed which represent the set of relations on X having a cycle of length 1, 3, 5 and 7, respectively. Then the intersection of $\overline{\text{cyc1}}$, $\overline{\text{cyc3}}$, $\overline{\text{cyc5}}$ and $\overline{\text{cyc7}}$ yielded the desired vector. The developments of cyc5 and especially of cyc7 are very technical and complex. For example, in case of cyc7 the relation-algebraic expression describes that for a given relation $R : X \leftrightarrow X$ there exist three pairs $(u_1, u_2) \in X \times X$, $(v_1, v_2) \in X \times X$ and $(w_1, w_2) \in X \times X$ and an element $x \in X$ such that the tuples (u_2, u_1, x, v_2, v_1) and (v_1, w_1, w_2, u_2) are paths in R and, therefore, the tuple $(u_2, u_1, x, v_2, v_1, w_1, w_2, u_2)$ is a cycle of length 7 in R .

A relation $R : X \leftrightarrow X$ has no cycle of length 1 if $R \subseteq \overline{\mathbf{I}}$, no cycle of length 3 if $R^3 \subseteq \overline{\mathbf{I}}$, no cycle of length 5 if $R^5 \subseteq \overline{\mathbf{I}}$ and no cycle of length 7 if $R^7 \subseteq \overline{\mathbf{I}}$, where powers are defined as usual. The method of this paper allows using these specifications directly leading to four vectors nocyc1 , nocyc3 , nocyc5 and nocyc7 of type $[X \leftrightarrow X] \leftrightarrow \mathbf{1}$ which represent the set of relations on X having no cycle of length 1, 3, 5 and 7, respectively. Then the intersection of nocyc1 , nocyc3 , nocyc5 and nocyc7 yields the vector we are looking for.

For the following, let an arbitrary $r : X \times X \leftrightarrow \mathbf{1}$ be given. With regard to the inclusion $R \subseteq \overline{\mathbf{I}}$ we have $\nu_r(R) = r$ and $\nu_r(\overline{\mathbf{I}}) = \text{vec}(\overline{\mathbf{I}}) = \overline{\text{vec}(\mathbf{I})}$, hence $\nu_r(R)[\mathbf{M}/r] = \mathbf{M}$ and $\nu_r(\overline{\mathbf{I}})[\mathbf{M}/r] = \text{vec}(\mathbf{I}); \mathbf{L} = \overline{\text{vec}(\mathbf{I}); \mathbf{L}}$ for the replacement of r by the membership relation $\mathbf{M} : X \times X \leftrightarrow [X \leftrightarrow X]$. So, Theorem 4.2 yields

$$\text{nocyc1} := \overline{\mathbf{L}; (\mathbf{M} \cap \text{vec}(\mathbf{I}); \mathbf{L})}^\top.$$

In case of the inclusions $R^3 \subseteq \overline{\mathbf{I}}$, $R^5 \subseteq \overline{\mathbf{I}}$ and $R^7 \subseteq \overline{\mathbf{I}}$ we work with the equivalent inclusions $(R^2)^\top \subseteq \overline{\mathbf{R}}$, $(R^3)^\top \subseteq \overline{\mathbf{R}^2}$ and $(R^4)^\top \subseteq \overline{\mathbf{R}^3}$ since these lead to more efficient RELVIEW-programs than the original inclusions. For $(R^2)^\top \subseteq \overline{\mathbf{R}}$ we get

$$\nu_r((R^2)^\top) = \mathbf{S}; \nu_r(R; R) = \mathbf{S}; \mathbf{C}^\top; \overline{[\nu_r(R), \nu_r(R)]} = \mathbf{S}; \mathbf{C}^\top; \overline{[r, r]}$$

and, hence, $\nu_r((R^2)^\top)[\mathbf{M}/r] = \mathbf{S}; \mathbf{C}^\top; \overline{[\mathbf{M}, \mathbf{M}]}$ for the left-hand-side, and

$$\nu_r(\overline{\mathbf{R}}) = \overline{\nu_r(R)} = \overline{\mathbf{r}}$$

and, hence, $\nu_r(\overline{R})[M/r] = \overline{M}$ for the right-hand-side. If we apply Theorem 4.2 to these results, we obtain the vector

$$nocyc3 := \overline{\mathbb{L}; (\mathbb{S}; \mathbb{C}^\top; \llbracket \mathbb{M}, \mathbb{M} \rrbracket \cap \mathbb{M})}^\top.$$

The remaining inclusions can be treated analogously. From $(R^3)^\top \subseteq \overline{R^2}$ we get

$$nocyc5 := \overline{\mathbb{L}; (\mathbb{S}; \mathbb{C}^\top; \llbracket \mathbb{M}, H \rrbracket \cap H)}^\top, \text{ where } H := \mathbb{C}^\top; \llbracket \mathbb{M}, \mathbb{M} \rrbracket,$$

and inclusion $(R^4)^\top \subseteq \overline{R^3}$ leads to

$$nocyc7 := \overline{\mathbb{L}; (\mathbb{S}; \mathbb{C}^\top; \llbracket H, H \rrbracket \cap \mathbb{C}^\top; \llbracket \mathbb{M}, H \rrbracket)}^\top, \text{ where } H := \mathbb{C}^\top; \llbracket \mathbb{M}, \mathbb{M} \rrbracket.$$

We have implemented the relation-algebraic specifications developed in this section in RELVIEW and have compared the running times with those given in [7]. Doing so, we have used the same environment as mentioned in [7], i.e., version 8.2 of RELVIEW² on a PC with 2 CPUs of type Intel[®] Xeon[®] E5-2698, each with 20 cores and 3.60 GHz base frequency, 512 GByte RAM and running Arch Linux 5.2.0. We only present the data for $|X| = 7$. Using the RELVIEW-programs resulting from [7], it takes 138.67s to compute the vector that represents the set of 188 553 949 010 868 relations on X which have a kernel and 32 220.55s to compute the vector that represents the set of 16 230 843 049 relations on X which satisfy Richardson's criterion. The RELVIEW-programs we have obtained from the specifications of this paper need 201.31s for computing the first vector and 18 843.34s for computing the second one.

From Lemma 3.2 we get $\nu_r(\mathfrak{R}; R; \mathfrak{S}) = (\mathfrak{R} \parallel \mathfrak{S}^\top); \nu_r(R)$ for all $R : X \leftrightarrow Y$ and $\mathfrak{R}, \mathfrak{S} \in \mathbf{RE}_R$, where $r := \text{vec}(R)$. In Sect. 3 we have mentioned that this often leads to more efficient implementations. If we proceed the above presented calculation for $\nu_r(\text{syq}(\overline{M}, R; \mathbb{M}))$ with $\overline{\nu_r(R; \mathbb{M})} = (\mathbb{I} \parallel \mathbb{M}^\top); \nu_r(R)$ and $\nu_r(\mathbb{M}^\top; R; \mathbb{M}) = (\mathbb{M}^\top \parallel \mathbb{M}^\top); \nu_r(R)$ after the third step, we obtain the variant

$$nokernel := \overline{\overline{\overline{\mathbb{L}; (\mathbb{C}^\top; \llbracket \overline{mt}; \mathbb{L}, (\mathbb{I} \parallel \mathbb{M}^\top); \mathbb{M} \rrbracket \cap (\mathbb{M}^\top \parallel \mathbb{M}^\top); \mathbb{M} \rrbracket \cap i; \mathbb{L})}}}}^\top$$

the RELVIEW-implementation of which allows to compute the vector representation of the set of relations on X having kernels for $|X| = 7$ in 189.89s.

6 Further Applications

We have applied our method to many other classes of specific relations. These include the remaining three criteria for the existence of kernels treated in [7], i.e., bipartite relations, progressively finite relations and symmetric and irreflexive relations. Also many of the vectors presented in [6] have been redeveloped using

² This is the newest version of the tool. It is described at the Web-site [14] and the source code is available from Github via [15] and from Zenodo via [16].

the new method. We also have applied the new method to classes of relations not treated so far, viz. lattices, bounded partial orders, finite directed acyclic graphs and arborescences, tournaments, rectangular relations, difunctional relations, general Ferrers relations, strongly connected relations and maps having fixpoints.

Sometimes the specification of a property $P(R)$ of a relation R leads to an inequation $\mathfrak{R} \neq \mathbf{O}$, with a relation-algebraic expression $\mathfrak{R} \in \mathbf{RE}_R$. An example is the specification of R to have a kernel by $\text{syq}(\overline{M}, R; M) \cap I \neq \mathbf{O}$ in Sect. 5. Also the specification of R to be a bounded partial order and of R to be a map with a fixed point leads to such inequations, viz. to $\overline{R}; L \neq \mathbf{O}$ (existence of a least element) and $\overline{R^T}; L \neq \mathbf{O}$ (existence of a greatest element) in the first case and to $R \cap I \neq \mathbf{O}$ (existence of a reflexive element) in the second case.

To cope with the inequation $\text{syq}(\overline{M}, R; M) \cap I \neq \mathbf{O}$, in Sect. 5 we consider the equation $\text{syq}(\overline{M}, R; M) \cap I = \mathbf{O}$ instead, which specifies R to have no kernel. Then we transform it into the equivalent inclusion $\text{syq}(\overline{M}, R; M) \subseteq \overline{I}$, apply our method to the latter and, finally, form the complement of the result to obtain the vector we are actually interested in. Since the carrier set of R is non-empty, $\text{syq}(\overline{M}, R; M) \cap I \neq \mathbf{O}$ holds iff $L \subseteq L; (\text{syq}(\overline{M}, R; M) \cap I); L$. We have applied our method also to that inclusion. But the corresponding RELVIEW-program proved to be less efficient than that we have obtained from the approach of Sect. 5.

That the inequation $\mathfrak{R} \neq \mathbf{O}$ is equivalent to the inclusion $L \subseteq L; \mathfrak{R}; L$ for relations on non-empty carrier sets we also have used to get vector representations of the set of bounded partial orders and of maps with fixed points. In each case the specific form of the expression \mathfrak{R} allows to apply Lemma 3.2 for computing $\nu_r(L; \mathfrak{R}; L)$. E.g., for the inclusion $L \subseteq L; (R \cap I); L$ we get $\nu_r(L) = L$ for the left-hand side and $\nu_r(L; (R \cap I); L) = (L \parallel L^T); \nu_r(R \cap I) = L; (r \cap \text{vec}(I))$ for the right-hand side. In combination with Theorem 4.2 this yields $L; \overline{L}; (\overline{M \cap \text{vec}(I)}; L)$ as vector representation of the set of relations with a reflexive element, which still can be simplified to $(L; (M \cap \text{vec}(I); L))^T$. With the RELVIEW-program obtained from the simplified specification and the vectors of [6] for the classes of univalent and total relations, respectively, we have been able to compute the vector representation of the set of maps on X having fixed points up to $|X| = 240$. For 240 elements 5.55 s are needed to get the result and to store it in a ROBDD with 229 439 vertices. Our RELVIEW-experiments show that the percentage of maps having fixed points decreases from 100% if $|X| = 1$ to 63.284 if $|X| = 240$. This is in accordance with the well-known result that, if P_n denotes the probability that a map on an n -element set selected uniformly at random has a fixed point, then P_n tends to $1 - \frac{1}{e}$ if n tends to infinity.

7 Concluding Remarks

The computational power obtained by the use of ROBDDs and RELVIEW becomes clear if we compare the running times mentioned in Sect. 5 and 6 with the times needed in case of a “classical” brute-force approach. E.g., if we assume that some algorithm could generate every map on a given finite set X and test

the existence of a fixed point in, say, 10^{-6} seconds, it would take $2.82 \cdot 10^{11}$ seconds (i.e., more than 9000 years) for this task already if $|X| = 15$, since in this case there are 282 325 794 823 047 151 maps on X having a fixed point. RELVIEW only needs 0.019s to compute a vector representing this set.

Membership relations $M : X \times Y \leftrightarrow [X \leftrightarrow Y]$ play a central role in our approach. The variable ordering used in RELVIEW allows to implement M by a ROBDD the number of vertices is linear in the size of $X \times Y$. Besides the very efficient ROBDD-implementation of the relational operations this specific feature of the tool seems to be the main reason for the amazing computational power in case of problems that deal with the computation of a subset of a powerset.

In addition to applying the theory to further examples and applications, there are at least two theoretical topics related to the material of this paper we would like to investigate in the future. The first topic is the question whether it is possible to find for any property $P(R)$ of relations $R : X \leftrightarrow Y$ expressed in second order logic an equivalent finite set of inclusions $\mathfrak{R}_i \subseteq \mathfrak{S}_i$ with $\mathfrak{R}_i, \mathfrak{S}_i \in \mathbf{RE}_R$ for all i . If this is the case, then the method of this paper could be applied immediately to the inclusions to get a vector that represents the set of all these relations. If this is not the case, it would be interesting to characterize the second order properties for which an equivalent description by a finite set of inclusions $\{\mathfrak{R}_1 \subseteq \mathfrak{S}_1, \dots, \mathfrak{R}_n \subseteq \mathfrak{S}_n\}$ is possible, i.e., for which our method is applicable. The second topic is the above mentioned conjecture about the probability of a relation selected uniformly at random from $[X \leftrightarrow X]$ to have a kernel. The conjecture states that this probability tends to zero if $|X|$ tends to infinity. So far we only have been able to prove that, given any (but fixed) $k \in \mathbb{N}$, the probability of a relation selected uniformly at random from $[X \leftrightarrow X]$ to have a kernel with at most k elements tends to zero if $|X|$ tends to infinity.

Acknowledgement. We thank the referees for their very valuable remarks.

References

1. Berghammer, R., Neumann, F.: RELVIEW – an OBDD-based computer algebra system for relations. In: Ganzha, V.G., Mayr, E.W., Vorozhtsov, E.V. (eds.) CASC 2005. LNCS, vol. 3718, pp. 40–51. Springer, Heidelberg (2005). https://doi.org/10.1007/11555964_4
2. Berghammer, R.: Applying relation algebra and REL VIEW to solve problems on orders and lattices. *Acta Infor.* **45**, 211–236 (2008)
3. Berghammer, R.: Relation-algebraic modeling and solution of chessboard independence and domination problems. *J. Logic and Alg. Progr.* **81**, 625–642 (2012)
4. Berghammer, R.: Column-wise extendible vector expressions and the relational computation of sets of sets. In: Hinze, R., Voigtländer, J. (eds.) MPC 2015. LNCS, vol. 9129, pp. 238–256. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-19797-5_12
5. Berghammer, R.: Tool-based relational investigation of closure-interior relatives for finite topological spaces. In: Höfner, P., Pous, D., Struth, G. (eds.) RAMICS 2017. LNCS, vol. 10226, pp. 60–76. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-57418-9_4

6. Berghammer, R.: Relational computation of sets of relations. In: Fahrenberg, U., Gehrke, M., Santocanale, L., Winter, M. (eds.) RAMiCS 2021. LNCS, vol. 13027, pp. 54–71. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-88701-8_4
7. Berghammer, R., Kulczynski, M.: Experimental investigation of sufficient criteria for relations to have kernels. In: Fahrenberg, U., Gehrke, M., Santocanale, L., Winter, M. (eds.) RAMiCS 2021. LNCS, vol. 13027, pp. 72–89. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-88701-8_5
8. Desharnais, J.: Monomorphic characterization of n -ary direct products. Inf. Sci. **119**(3–4), 275–288 (1999)
9. Kehden, B.: Evaluating sets of search points using relational algebra. In: Schmidt, R.A. (ed.) RelMiCS 2006. LNCS, vol. 4136, pp. 266–280. Springer, Heidelberg (2006). https://doi.org/10.1007/11828563_18
10. Schmidt, G., Ströhlein, T.: Relations and Graphs. Springer, Heidelberg (1993)
11. Schmidt, G.: Relational Mathematics. Cambridge University Press, Cambridge (2010)
12. von Neumann, J., Morgenstern, O.: Theory of Games and Economic Behaviour. Princeton University Press, Princeton (1944)
13. Winter, M.: Goguen Categories - A Categorical Approach to L -Fuzzy Relations. Trends in Logic, vol. 25. Springer, Heidelberg (2007). <https://doi.org/10.1007/978-1-4020-6164-6>
14. <https://www.rpe.informatik.uni-kiel.de/en/research/relview>
15. <https://github.com/relview>
16. <https://zenodo.org/record/4708085#.YICAmS0RppR>